ISP RAS
Institute for System Programming
of the Russian Academy of Sciences

The ISP RAS Effort to Improve GCC for Itanium

Arutyun I. Avetisyan

Gelato ICE, April 24, 2006

The ISP RAS

- One of the leading IT organizations of RAS
- Counts about 200 highly qualified researchers and software engineers
  - 12 Professors and 43 Ph.D.'s
- Leads the Departments of System Programming in Moscow State University and Moscow Institute for Physics and Technology
- Funded by RAS and Russian research foundations
- Cooperates with many scientific research institutions
  - Fraunhofer (Germany), NPS (USA), RAL (UK), INRIA (France)
- Works as a contractor with leading industry companies
  - Intel Corp, HP Company, Telelogic AB
    VIA Technologies, Nortel Networks, and others

Background in compilers

- 1980th – development and implementation of system software for the “Elektronika SS-BIS” supercomputer (similar to Cray)
- 1990th – design and implementation of the SQL compiler and request optimizer for the free SQL server supported by the Free Software Foundation
- 1990th – development of the Protel and Protel 2 optimizing compilers for HP-UX against contract with Nortel Networks
- 2000th – development of Integrated Research Environment (IRE) for program analysis and optimization

Presentation agenda

- Workflow of the ISP RAS GCC activity
- Past and present GCC projects
- Other compiler works of the ISP RAS
- Future GCC activities and collaboration
- Gelato GCC Workshop in Moscow

Workflow of our GCC activity

- The ISP RAS is working on GCC under contract with HP Company
- Our primary goal is improving GCC on Itanium Processor Family
- We work in contact with GCC developers and with engineers from HP, Intel, and Red Hat
- Results of our work are contributed back to the GCC community

GCC contribution cycle

- Create and test your patches
  - using a variety of platforms (x86, x86_64, ia64, powerpc)
- Sort out intellectual property issues
  - sign FSF copyright papers
- Send your patches for a public review
- Collaborate with other developers
  - Participate in development mailing lists
  - Make a presentation on the GCC Summit
- Support your code
  - Fix the bugs you've introduced or exposed
  - Apply for a SVN write access

More at http://gcc.gnu.org/contribute.html

Completed work

- Add speculation support for IA-64 to the GCC instruction scheduler
  - done under 6 months contract with HP with a team of three
  - 6 large patches and a number of minor fixes are approved
  - committed to GCC mainline in March 2006
  - will be included in GCC 4.2 release
  - gives ~0.5% improvement on SPEC INT 2000 (up to 6% on selected tests) and ~1.4% improvement on SPEC FP 2000 (up to 11.7% on selected tests)
- Propagate alias information from the Tree SSA to the RTL level
  - done under the same contract
  - will be included in the ia64-improvements GCC branch

Current GCC project

- Implement new aggressive interblock instruction scheduler for GCC
- Choose *selective scheduling* approach as a start
  - focuses on VLIW architectures, but general enough for others
  - supports instruction cloning, speculative code motion, multiway branching, register renaming, forward substitution
  - percolation scheduling and resource-constrained software pipelining ideas are also used
- Submit and support the speculation patches
- 15 months project, started September 2005
- Planned for a team of four
- Consulting is provided by Vladimir N. Makarov, Red Hat

Timeline of our GCC work

2005:

- January : start of IA-64 speculation project
- April : preliminary report on Gelato meeting
- June : a presentation on GCC Summit 2005
- September : start of new scheduler work
- December : speculation patches are posted on the gcc-patches list

2006:

- March : speculation patches are approved
- April : Gelato ICE talk on the scheduler project
- June* : a presentation on GCC Summit 2006
- August* : new branch in the FSF repository
- December* : the scheduler project completion

Early 2007* : the scheduler patches are ready for GCC mainline

ISP RAS compiler projects

- Integrated research environment (IRE)
- Interprocedural and whole-program optimizations
- Static checkers
- Source code obfuscation
- Cross development tools
- Porting GCC to embedded platforms

Integrated Research Environment

- Launched in 2001 under contract with Intel Corp.
- Designed for quick prototyping and evaluating of compiler algorithms and static analyses
- Defines a common intermediate representation
  - medium-level 3-address code, infinite virtual register set
  - language-independent
- Has a C language front-end, working on C++
- Implements a number of analyses and optimizations
  - IR / CFG / DDG/ dominators manipulation, SSA form
  - dataflow analysis (reaching definitions, alias analysis)
  - classic optimizations, interprocedural framework
  - profiling/slicing/execution traces

Past and present IRE-based projects

- Program obfuscation and deobfuscation
  - evaluation of a proprietary obfuscation algorithm under contract with **Cloakware Corp. (Canada)**, 2002
  - an environment for obfuscation of integrated circuit design under contract with **Zelenograd ZIC (Russia)**, 2005-present
- Research on profile-based and whole-program optimizations
  - supported by **RFBR (Russia),** 2002
  - supported by **Academy of Sciences (Russia),** 2003-2005
- Development of SVaCE Detector tool for finding security vulnerabilities and critical errors in C/C++ source code
  - under contract with **Nortel Networks (Canada)**, 2003-2004
  - under contract with **the ROSNAUKA (Russia)**, 2005-present

SVaCE Detector Features

- Detects security vulnerabilities
  - buffer overflow, format string
  - access beyond the object bounds
- Detects critical errors
  - null pointer dereferences
  - use after free/double free/memory leaks
- Interprocedural level of source code analysis
- Uses heuristics for reducing the amount of false warnings
  - uses backward analysis for detecting source reasons for each warning and severance of false warnings

Results and Free Trial

The following table shows the evaluation of the SVaCE Detector on a three small open source programs:

	Bftpd 1.0.24	Surfboard 1.1.8	Muh 2.05d
Source LOCs	3326	675	3228
Functions in project	114	18	95
Analysis time,sec.*	30.41	5.82	55.77
Memory usage	23 MB	4.6 MB	28.4 MB
Total warnings	235	52	129
True positives	94(40,0%)	30(57,7%)	112(86,6%)

If you are interested, we offer you to evaluate SVaCE Detector on an arbitrary source code base.

You can find additional information about terms and conditions of Free Trial at:

www.ispras.ru/groups/ctt/svace.html

Proposed activities and collaboration

- Participate in development and implementation of link-time optimizations (LTO) for GCC
- Participate in an implementation of memory hierarchy optimizations for GCC
- Implement static checkers for production compilers
  - security vulnerabilities
  - critical errors
- Implement other static analysis tools
  - understanding program semantics
  - concept validation

Linux Verification Center

- In 2005, ISP RAS has won a federal tender from Russian Federal Agency for Science and Innovations to establish **Linux Verification Center** in Russia (http://linuxtesting.org)
- The current flagship project of the Center is **Open Linux VERification (OLVER)**. The purpose of the project is to supplement **Linux Standard Base (LSB)** standard with formal specifications and corresponding conformance and functional tests
- The OLVER project is coordinated with the Free Standards Group and the Austin Group (maintainers of LSB and SUS / POSIX standards)

Gelato GCC Workshop in Moscow

- Moscow, Russia, August 2006
- Organized to bring together GCC developers and Gelato members interested in improving GCC for Itanium
- Aims to review the goals of Gelato GCC group established on Geneva Workshop in January 2005 and discuss other beneficial projects
- Intel Corp. and HP Company will be official sponsors of the Workshop

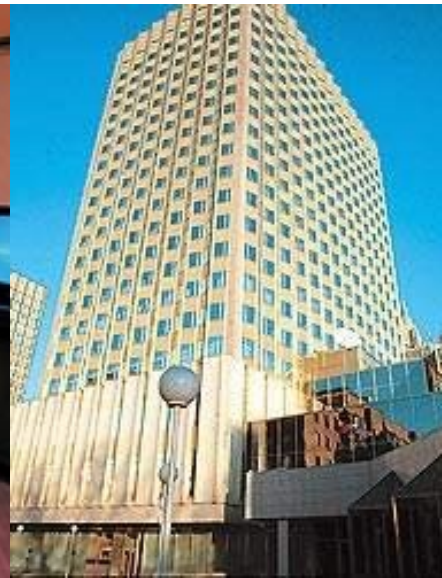
Proposed agenda

- Reviewing the topics discussed in Geneva
  - memory disambiguation/array data dependence
  - modulo scheduling/rotating registers
  - instruction/superblock scheduling
- Presenting the work in progress
  - instruction scheduling/software pipelining
- Discussing other projects of high interest
  - interprocedural optimizations
  - memory hierarchy optimizations
  - parallelization and OpenMP
  - alternative backends for GCC
- Proposing a plan for the next year work of the group

Travel arrangements

- A flight is approx. 4 hrs from Europe, 11hrs from United States, 8 hrs from China
- A visa is required
- Typical weather is +20-25°C
- A 4*-5* hotel in the center of Moscow will be booked





Daily meetings

Conference halls of
Presidium Building of Russian Academy of Sciences

Social events

The Moscow Kremlin

Social events

The Armoury Chamber

…And a lot more to see in Moscow!